

Our Ref: OP1480-US

Prior Art Reference:

Japanese Patent Laid-Open Publication No. Hei 4-181282
Date of Laid-Open: June 29, 1992
Title: ENCRYPTION SYSTEM FOR FILE
Patent Application No. Hei 2-308893
Filing Date: November 16, 1990
Inventor: Yasuhiro ISHII
c/o Kanagawa Plant of Kabushiki Kaisha Hitachi
Seisakusho
Hatano-shi, Kanagawa-ken, Japan
Applicant: KABUSHIKI KAISHA HITACHI SEISAKUSHO
Chiyoda-ku, Tokyo, Japan

PARTIAL TRANSLATION:

[Technical Field of the Invention]

The present invention relates to a method of storing files in a computer, more particularly, to a method of storing a data in an encrypted form in another computer connected via a communication line.

[Prior Art]

Regarding the conventional encryption system, it is discussed on pages 276-306 of New Development of Computer Data Protection.

According to this, a line encryption has an encryption key which is common to intra-communication computers, and a data to be transmitted via the line is encrypted according to that encryption key, and the receiving side decrypts the encryption similarly by the encryption key.

For a file encryption, a file key corresponding to the file is generated, and the data in the file is encrypted/decrypted according to that key.

In order to safely store the data of one computer in a file of the other computer, the following processes are required.

First, a communication in an encryption form is made, by utilizing a line encryption procedure, between a computer which is the source of generating an encryption and a computer wherein the data is stored, thereby to transfer the data safely. Next, the data is encrypted by utilizing a file encryption procedure and stored in the file.

[Problems to be Solved by the Invention]

The above-discussed prior art did not give any consideration to an encryption system for a file control wherein a file is accessed via a communication line such as a file server system, and, therefore, it had the following problems.

(1) In order to protect the data on the line, the line encryption is made every time when the data is transmitted from a work station to a file server or from the file server to the work station. Further, for the purpose of file protection, it is necessary to perform the file encryption every time when the data is stored or the data is read in the file server. Thus, it requires to do the line encryption and the file encryption doubly, the processing efficiency is poor.

(2) The encryption key management must be carried out strictly by a system manager as it is a matter of secret. However, from the standpoint of the utilization mode for utilizing the file server, it cannot be expected that the user manages the key strictly. Thus, it is required to simplify the key manager.

The present invention was made to solve such problems.

An object of the present invention is to perform the encryption process of the file data efficiently, and, at the same time, to provide a simple method of managing an encryption key.

/ / / / / / / LAST ITEM / / / / / / / /

⑫ 公開特許公報(A) 平4-181282

⑬ Int. Cl.⁵

G 09 C 1/00
G 06 F 12/00
H 04 L 9/00
9/10
9/12

識別記号

5 3 7 H

庁内整理番号

7922-5L
8944-5B

⑭ 公開 平成4年(1992)6月29日

7117-5K H 04 L 9/00

Z

審査請求 未請求 請求項の数 3 (全6頁)

⑮ 発明の名称 ファイルの暗号方式

⑯ 特 願 平2-308893

⑰ 出 願 平2(1990)11月16日

⑱ 発 明 者 石 井 保 弘 神奈川県秦野市堀山下1番地 株式会社日立製作所神奈川工場内

⑲ 出 願 人 株式会社日立製作所 東京都千代田区神田駿河台4丁目6番地

⑳ 代 理 人 弁理士 小川 勝男 外1名

明 細 書

1. 発明の名称

ファイルの暗号方式

2. 特許請求の範囲

1. 複数の電子計算機を通信回線等を用いて接続し、各電子計算機で作成したファイルを接続された任意の電子計算機に保管するシステムにおいて、データ作成元の電子計算機側でファイルに保管するデータを暗号化し、該暗号化されたデータを通信回線等を使用して保管する電子計算機に送り、保管する電子計算機では暗号化されたデータをファイルとして保管することを特徴とするファイルの暗号方式。

2. 請求項(1)において、該保管先の電子計算機は通信回線等を使用して該暗号化されたデータをデータ作成元の各電子計算機に送り、各電子計算機はデータを復号化してファイルの元の内容を得ることを特徴とするファイルの暗号方式。

3. 請求項(1)又は(2)において、データ暗

号鍵をファイル使用者の暗号鍵にて暗号化して暗号化されたデータとともに保管先の電子計算機にて保管することを特徴とするファイルの暗号方式。

3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は電子計算機のファイルの保管方法に関し、特に、通信回線で接続された別の電子計算機にデータを暗号化して保管する方法に関する。

〔従来の技術〕

従来の暗号方式については暗号(コンピュータ・データ保護の新展開)第276ページから第306ページにおいて論じられている。

これによれば、回線暗号は通信する電子計算機同士が共通の暗号鍵を有し、この暗号鍵に従って回線に送出するデータを暗号化し、受信側は該暗号鍵により同様に復号化することになっている。

ファイル暗号はファイル対応にファイル鍵を生成し、この鍵に従ってファイル内データを暗号/復号化することになっている。

ある電子計算機上のデータを他の電子計算機のファイルに安全に格納するためには次の処理が必要である。まず、回線暗号手順を用いて作成元電子計算機と格納先電子計算機間で暗号通信を行い、データを安全に転送する。次に、ファイル暗号手順を用いて、データを暗号化してファイルに格納することとなる。

〔発明が解決しようとする課題〕

上記従来技術は、ファイルサーバ方式などのような通信回線を介してファイルをアクセスするファイル制御の暗号方式について配慮されておらず、次のような問題点があった。

(1) 回線上のデータ保護のために、ワークステーションからファイルサーバあるいはファイルサーバからワークステーションへのデータ送信の度に回線暗号を行い、また、ファイル保護のために、ファイルサーバにおいてファイルのデータの格納あるいはデータの読みだしの度にファイル暗号を行う必要がある。このように、回線暗号とファイル暗号を重複して行う必要があり、処理効率が悪

これによれば、ワークステーションからファイルサーバへファイルを格納する場合、各ワークステーションは格納したいデータを自ワークステーション内で作成した暗号鍵で暗号化してファイルサーバに送信し、ファイルサーバは暗号化されたデータをそのままファイルに書き込む。

また、ファイルサーバからデータを読み取る場合、ファイルサーバは暗号化されたデータをファイルから読み込み、これをそのままワークステーションに送る。ワークステーションは自ワークステーション内で管理している暗号鍵で復号化し、生のデータを得る。

ゆえに、ワークステーションからファイルサーバあるいはファイルサーバからワークステーション間の送信データは暗号化されており、回線上の機密を保つことができる。また、ファイルサーバのファイル内に格納されたデータも暗号化されており、ファイル上の機密を保つことができる。

このことより、ファイルサーバは復号/暗号処理をする必要がないので効率良く処理することが

(2) い。

(2) 暗号鍵管理は機密上、システム管理者がファイルサーバ上で厳格に行う必要がある。しかし、ファイルサーバの利用の利用形態からみて鍵をユーザが厳格に行うことは期待できない。故に、鍵管理者を簡素化する必要がある。

本発明は、このような問題点を解決するためになされたものである。

本発明の目的は、ファイルデータの暗号処理を効率よく行うとともに、簡素な暗号鍵管理方法を提供することにある。

〔課題を解決するための手段〕

上記の目的を達成するために、各ワークステーションのみがファイルデータを暗号/復号化し、ファイルサーバは暗号化されたデータを直接ファイルに書き込み、あるいは、読みだしするようにしたものである。

また、暗号鍵管理も各ワークステーションで行い、管理を局所化したものである。

〔作用〕

できる。また、ファイルサーバは、暗号を行わないので暗号鍵の管理は不要であり、鍵管理がワークステーション内で閉じるので安全性が高まるとともに処理を簡素化することができる。

〔実施例〕

以下、本発明の一実施例を第1図、第2図により説明する。

第2図に電子計算機の接続図を示す。ファイルサーバ1は実ディスク3を有し、LAN網2に接続されている。ワークステーション10-15も同じLAN網に接続されており、ファイルサーバ1と各ワークステーション10-15間は自由に通信できるようになっている。

第1図にファイルサーバ1とワークステーション10の処理ブロック図を示す。(ワークステーション11-15はワークステーション10と同一なのでここでは省略する。) ファイルサーバ1内には通信制御モジュール(CCM-S)101とファイル転送制御モジュール(FTM-S)102、ファイル制御モジュール(FCM-S)1

03があり、実ディスク3と接続されている。ワークステーション10内にはアプリケーションプログラム(AP-W)201とファイル制御モジュール(FCM-W)202、ファイル転送制御モジュール(FTM-W)203、および通信制御モジュール(CCM-W)204からなり、ファイル転送制御モジュール(FTM-W)203内は、暗号化ルーチン221と復号化ルーチン222、鍵管理ルーチン223、および、暗号鍵224からなる。ファイルサーバ1とワークステーション10はLAN網2にて接続されている。

次に書き込み時の処理手順について第3図を用いて説明する。

step301: アプリケーションプログラム(AP-W)201はファイル制御モジュール(FCM-W)202に対してライトモードでファイルのオープンを指示する。
step302: ファイル制御モジュール(FCM-W)202は仮想ディスク210上に仮想ファイルをアロケーションする。

204に渡す。

step309: 通信制御モジュール(CCM-W)204は暗号化されたデータをファイルサーバ1に送る。

step310: 通信制御モジュール(CCM-S)101は暗号化されたデータを受け取り、ファイル転送制御モジュール(FTM-S)102に渡す。

step311: ファイル転送制御モジュール(FTM-S)102はファイル制御モジュール(FCM-S)103に対してファイルのアロケーションを指示する。

step312: ファイル制御モジュール(FCM-S)103は実ディスク3上にファイルをアロケーションする。

step313: ファイル転送制御モジュール(FTM-S)102はファイル制御モジュール(FCM-S)103に対して暗号化されたデータの書き込みを指示する。

step314: ファイル制御モジュール(FCM-S)103は実ディスク3上に暗号化されたデータを

(3) step303: アプリケーションプログラム(AP-W)201はファイル制御モジュール(FCM-W)202に対してデータの書き込みを指示する。

step304: ファイル制御モジュール(FCM-W)202は仮想ディスク上にデータを書き込む。

step305: アプリケーションプログラム(AP-W)201はファイル制御モジュール(FCM-W)202に対してファイルのクローズを指示する。

step306: ファイル制御モジュール(FCM-W)202はファイル転送制御モジュール(FTM-W)203に対して仮想ファイルをファイルサーバに転送することを要求する。

step307: ファイル転送制御モジュール(FTM-W)203は鍵管理ルーチン223でファイルの暗号鍵を作成する。

step308: ファイルの転送制御モジュール(FTM-W)203は仮想ディスク210上の仮想ファイルのデータを読み取り、暗号化ルーチン221で暗号し、通信制御モジュール(CCM-W)

書き込む。

これにより、ファイルサーバ1の実ディスクには暗号化されたデータが格納される。また、ワークステーション10からファイルサーバ1へのデータ転送もデータが暗号化されているので盗聴などに対して安全である。

第4図にファイル読み取り処理手順について示す。

step401: アプリケーションプログラム(AP-W)201はファイル制御モジュール(FCM-W)202に対してリードモードでファイルをオープンする。

step402: ファイル制御モジュール(FCM-W)202はファイル転送制御モジュール(FTM-W)203に対してファイルサーバ1からのファイル転送を要求する。

step403: ファイル転送制御モジュール(FTM-W)203は通信制御モジュール(CCM-W)204および通信制御モジュール(CCM-S)101を介して、ファイル転送制御モジュール

(F T M - S) 1 0 2 に実ファイルの転送を要求する。

step404: ファイル転送制御モジュール (F T M - S) 1 0 2 はファイル制御モジュール (F C M - S) 1 0 3 に対してファイルの読み取りを指示する。

step405: ファイル制御モジュール (F C M - S) 1 0 3 は実ディスク 3 上の暗号化されたデータを読み取る。

step406: ファイル転送制御モジュール (F T M - S) 1 0 2 は通信制御モジュール (C C M - S) 1 0 1 に対して暗号化されたデータの転送を指示する。

step407: 通信制御モジュール (C C M - S) 1 0 1 はワークステーション 1 0 に暗号化されたデータを送信する。

step408: 通信制御モジュール (C C M - W) 2 0 4 は暗号化されたデータを受け取り、ファイル転送制御モジュール (F T M - W) 2 0 3 に渡す。

step409: ファイル転送制御モジュール (F T M

ファイルサーバ 1 の実ディスク 3 に格納されたデータを生のデータとして読み取ることができる。また、ファイルサーバ 1 からワークステーション 1 0 へのデータ転送もデータが暗号化されているので盗聴などに対して安全である。

また、鍵管理ルーチン 2 2 3 で生成した暗号鍵 2 2 4 はファイル所有者のマスタ鍵で暗号化し、ファイルのヘッダとしてデータに添付し、実ファイル 3 に格納しておく。これにより、ファイルの読みだし時、ヘッダの暗号された鍵を復号化し、この暗号鍵 2 2 4 でデータを復号化することができるので、鍵の管理をより簡単に済ませることができる。

このように、本実施例によれば次の効果がある。

(1) 1 回の暗号処理で L A N 網 2 上のデータの暗号化と、ファイルサーバ 1 の実ディスク 3 上のデータの暗号化が可能であり、処理効率を高めることができる。

(2) ワークステーション 1 0 でのみ暗号処理を行い、ファイルサーバ 1 では暗号処理を行わない。

(4) - W) 2 0 3 は鍵管理ルーチン 2 2 3 で暗号鍵 2 2 4 を設定する。

step410: ファイル転送制御モジュール (F T M - W) 2 0 3 は暗号化データを復号化ルーチン 2 2 2 で元のデータに復元し、仮想ディスク 2 1 0 に元のデータを書き込む。

step411: アプリケーションプログラム (A P - W) 2 0 1 はファイル制御モジュール (F C M - W) 2 0 2 に対してファイルの読み取りを指示する。

step412: ファイル制御モジュール (F C M - W) 2 0 2 は仮想ディスク上のデータを読み取る。

step413: アプリケーションプログラム (A P - W) 2 0 1 はファイル制御モジュール (F C M - W) 2 0 2 に対してファイルのクローズを指示する。

step414: ファイル制御モジュール (F C M - W) 2 0 2 は仮想ディスク上のファイルを消去する。

これにより、ワークステーション 1 0 上のアプリケーションプログラム (A P - W) 2 0 1 はフ

ゆえに、鍵管理はワークステーション 1 0 ないに留めることができるので、鍵管理が飛躍的に簡単となる。

〔発明の効果〕

本発明に依れば、次の効果がある。

(1) 回線上のデータとファイル上のデータを同一の暗号化データとするので、回線暗号とファイル暗号を 1 階の暗号処理で済ませることができるので、暗号処理効率を高めることができる。

(2) ファイル作成元でのみ暗号処理を行い、ファイル格納先では暗号処理を行わない。ゆえに、鍵管理を局所化できるので、鍵管理が飛躍的に簡単となる。

4. 図面の簡単な説明

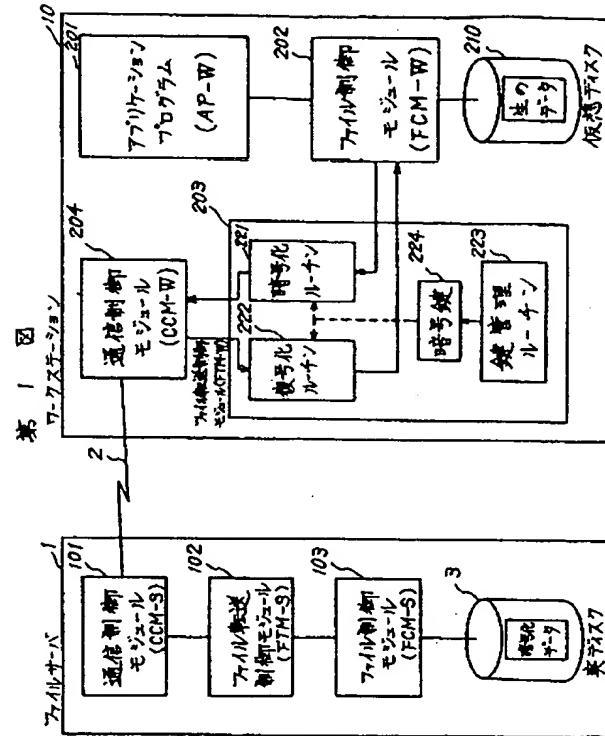
第 1 図は本発明の一実施例であるシステムの処理ブロック図、第 2 図はシステムの構成図、第 3 図はファイル書き込み時の処理フロー図、第 4 図はファイル読み取り時の処理フロー図である。

〔符号の説明〕

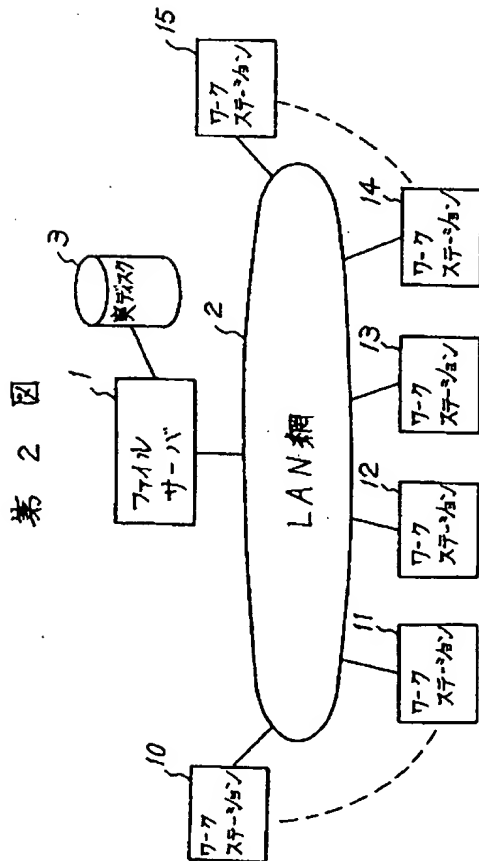
1 … ファイルサーバ、2 … L A N 網、3 … 実ディ

スク、10、11、12、13、14、15…ワークステーション、101…通信制御モジュール (CCM-S)、102…ファイル転送制御モジュール (FTM-S)、103…ファイル制御モジュール (FCCM-S)、201…アプリケーションプログラム (AP-W)、202…ファイル制御モジュール (FCM-W)、203…ファイル転送制御モジュール (FTM-W)、204…通信制御モジュール (CCM-W)、221…暗号化ルーチン、222…復号化ルーチン、223…鍵管理ルーチン、224…暗号鍵。

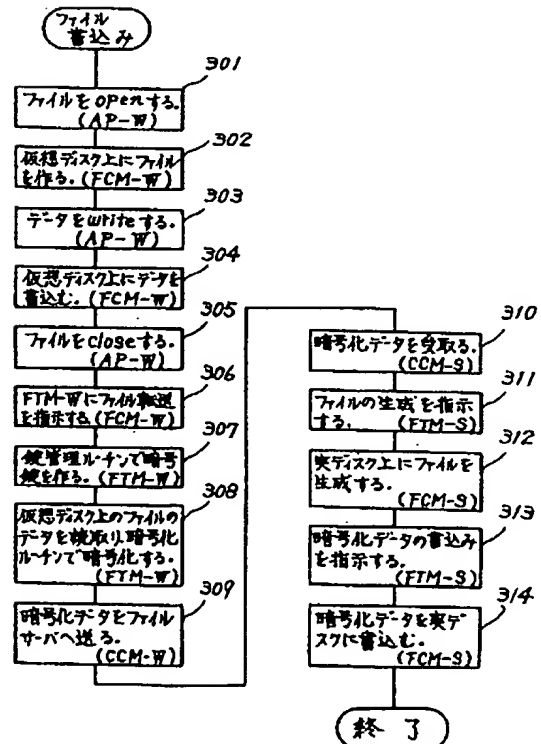
(5)



代理人弁護士 小川 勝 男



第3図



(6)

第 4 図

